Networking Algorithms

Mani Srivastava
UCLA

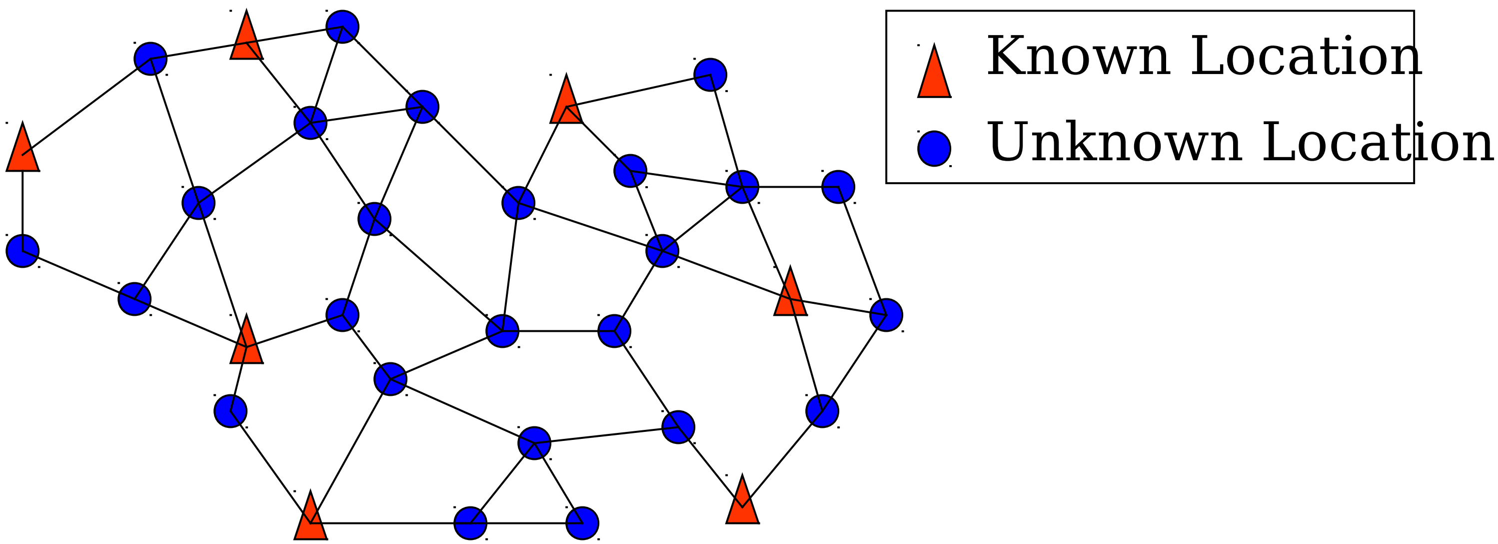
[Project: Dynamic Sensor Nets (ISI-East)]

Outline

I. Dynamic location discovery
II. Topology management
III. Dynamic MAC address assignment

I. Dynamic Location Discovery

- **Discovery of absolute and relative location important**
  - **Location-based naming and addressing, geographical routing, tracking**
- **GPS not enough**
  - **LOS-requirements, costly, large, power-hungry**
- **Ad hoc precludes trilateration with special high power beacons**
  - **also, susceptible to failure**
- **Problem: given a network of sensor nodes where a few nodes know their location (e.g. through GPS) how do we calculate the location of the other nodes?**

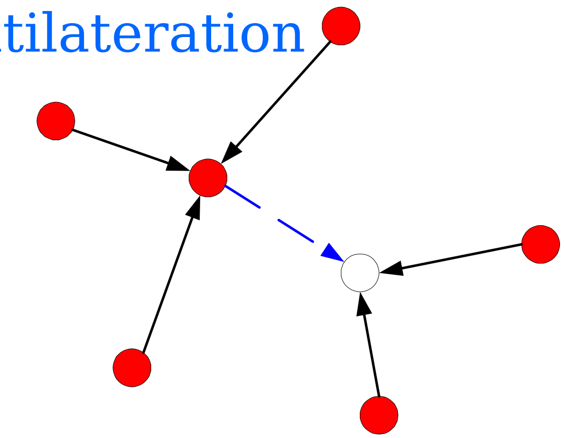


Ad-Hoc Localization System (AHLoS)

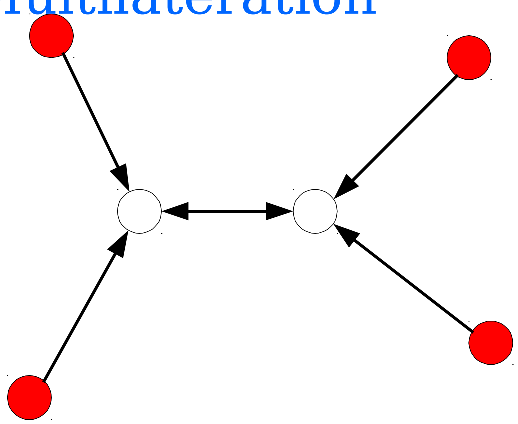
GOALS

- **Localization in a distributed fashion**
- **Trade-offs**
  - **Robustness**
  - **Computation vs. communication**
- **Ranging using Ultrasound**
- **Integrated with routing messages**
  - **Location discovery almost free**
- **Implementation**
  - **Ranging using radio-synchronized ultrasound**
    - **3m range, noisy**
  - **Accuracy:**
    - **Iterative: ~ 10 cm**
    - **Collaborative: ~ 3 cm**

Iterative Weighted Multilateration

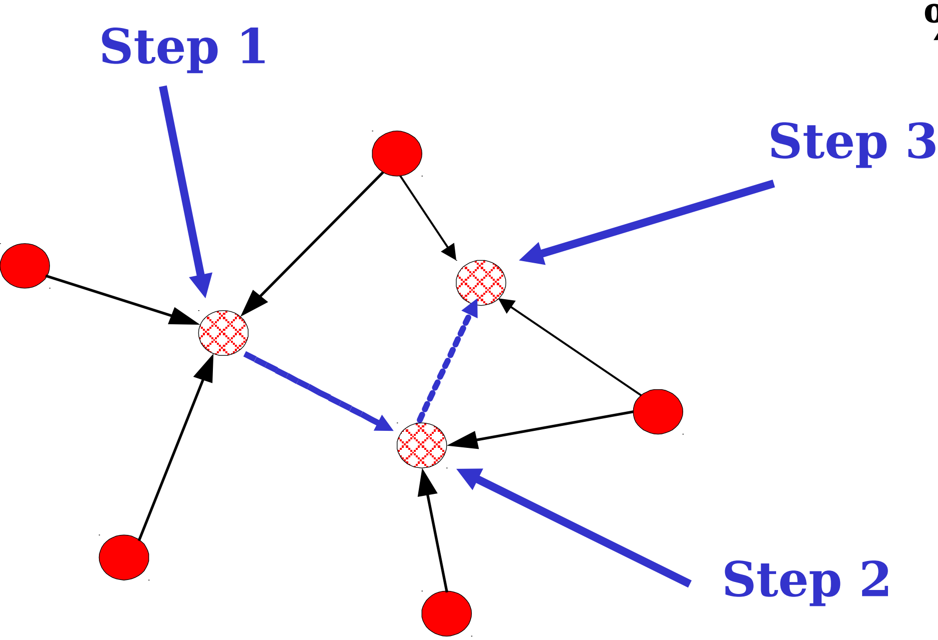


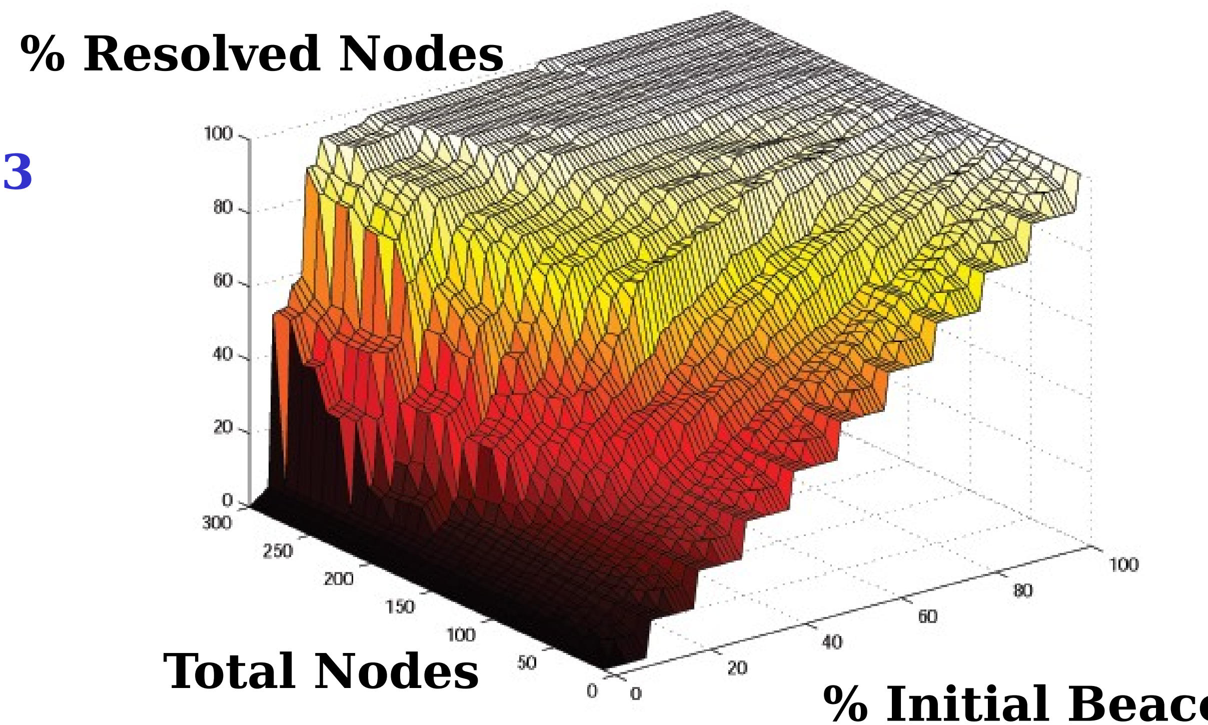
Collaborative Multilateration



Iterative Multilateration

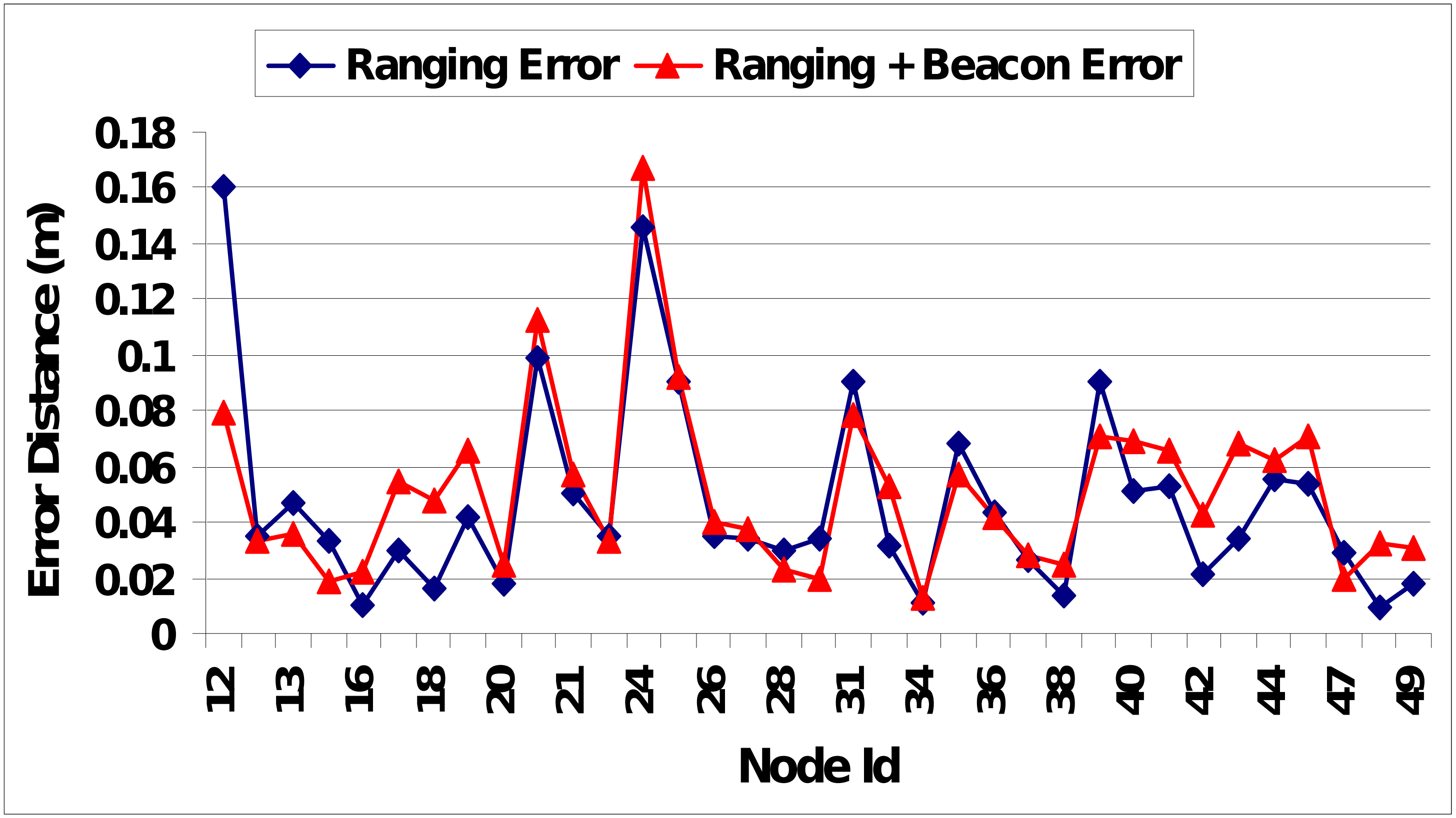
- **Atomic multilateration applied iteratively across the network**
  - **may stall if network is sparse, % of beacons is low, terrain obstacles**





Uniformly distributed deployment in a field 100x100. Node range = 10.

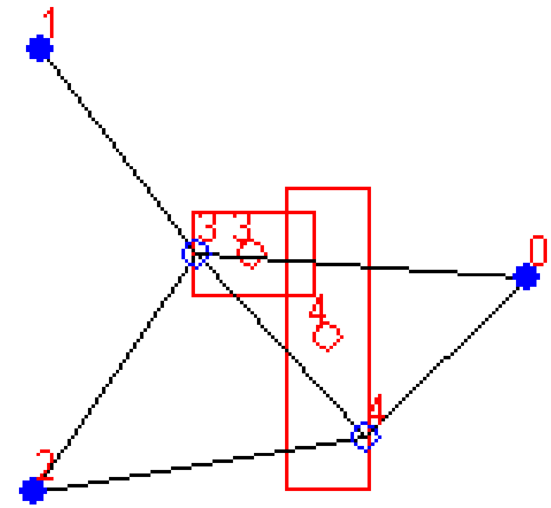
Iterative Multilateration Accuracy

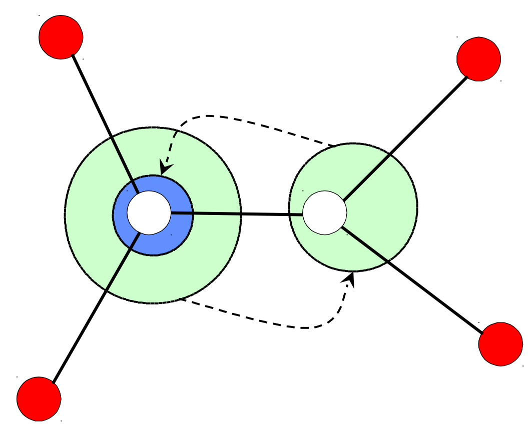


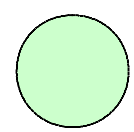
50 Nodes 10% beacons 20mm white gaussian ranging error

Collaborative Multilateration

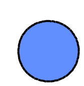
- **Step 1: form “Collaborative Subtrees” within their neighborhood**
  - **an unknown node is collaborative if it has at least 3 participating neighbors**
  - **a node is participating if it is either a beacon, or if it is an unknown node that is also participating**
  - **at each node at least one of its participating nodes are new to the set**
  - **at least one of the beacons used to determine the position of a node should not be collinear with the other beacons used to determine the node position**
- **Step II: obtain initial location estimate for subsequent computation**
  - **use beacon locations & hop distances to obtain approximate location bounds**
- **Step III: perform computation**
  - **Measurement Update part of Kalman Filter**
  - **Centralized, at a leader elected in the subtree**
  - **or, Fully Distributed**
    - **can start computing locations based on node connectivity and initial estimate**





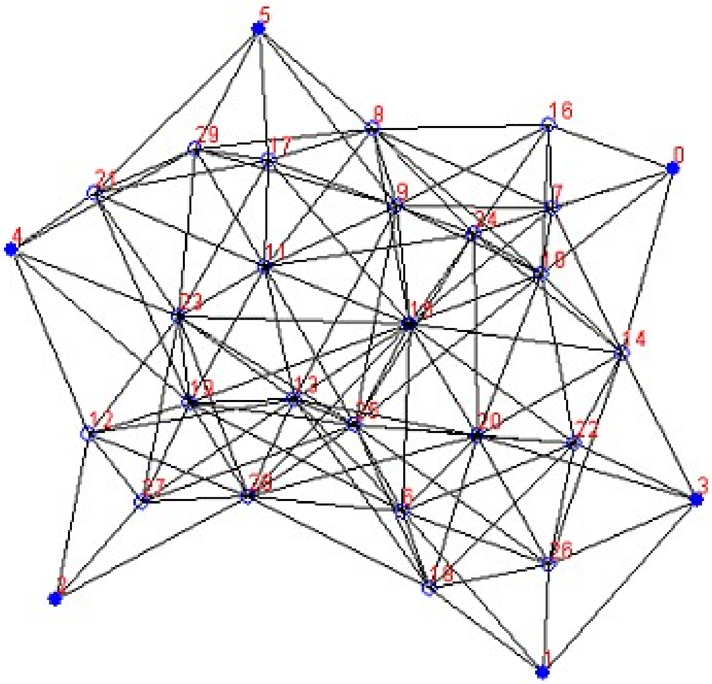


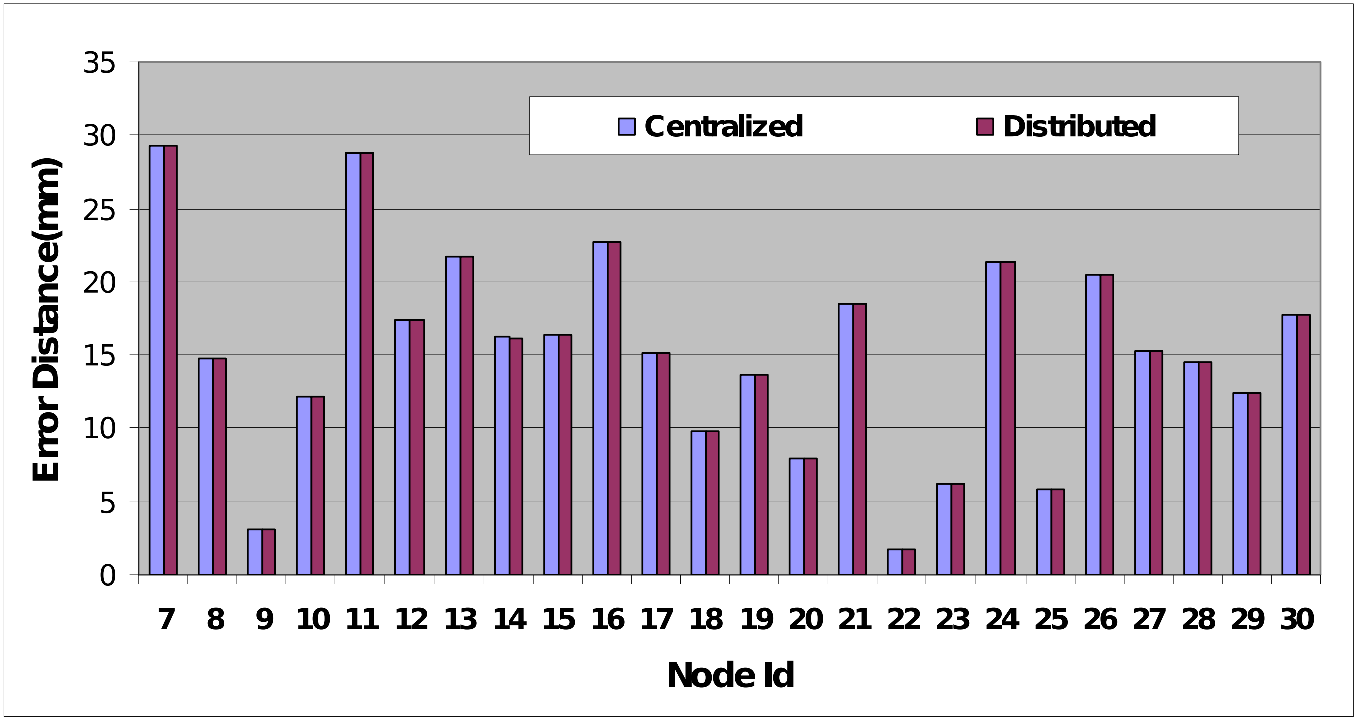
Uncertainty of estimat
location in first iteratio



Uncertainty of estimat
location in second iter

Example



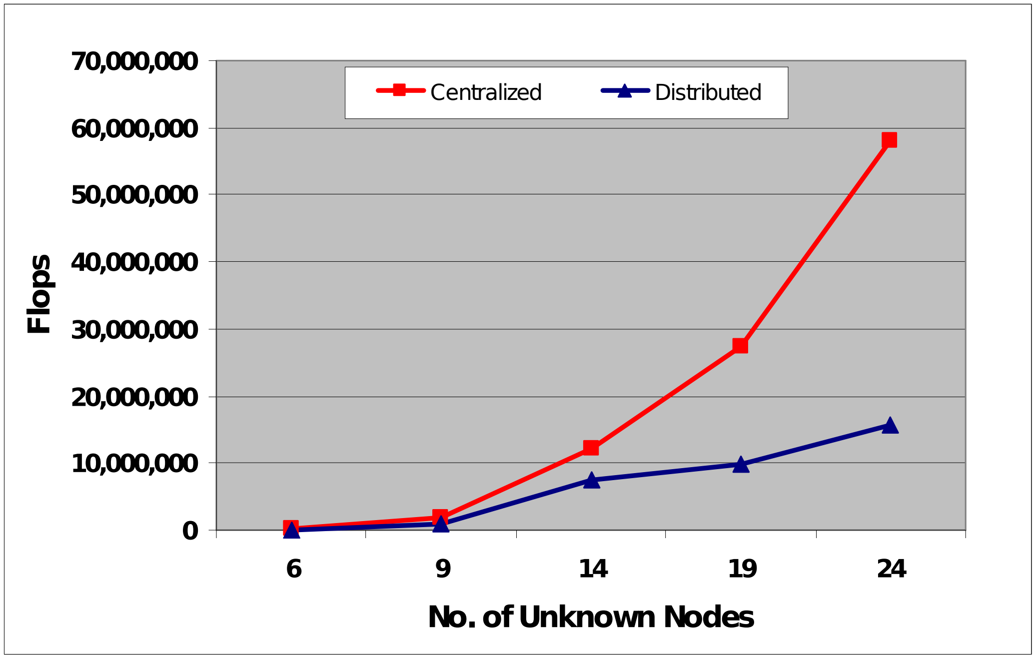


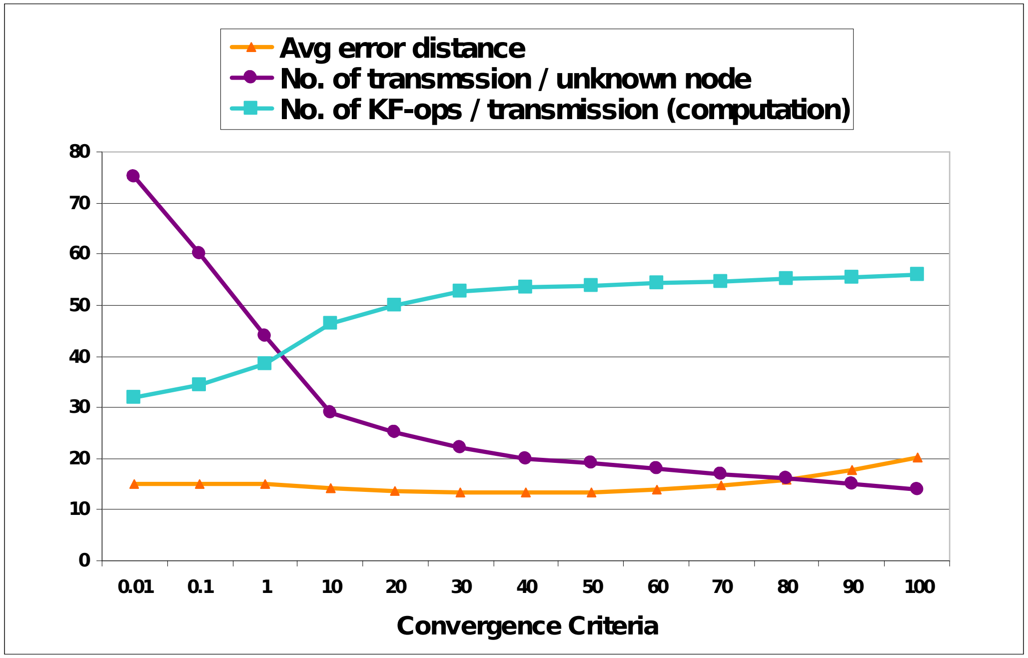
Network of 30 nodes

6 beacons, 24 unknowns

Ranging noise experimentally derived

Computation & Communication Expense





Computation Expense **Computation vs. Communication Tradeoff**

Results using the FLOPS command in MATLAB

- **Total number of transmissions:**
  - **Centralized 70 packets**
  - **Distributed 416 packets**
- **Centralized approach has additional overhead for leader election**

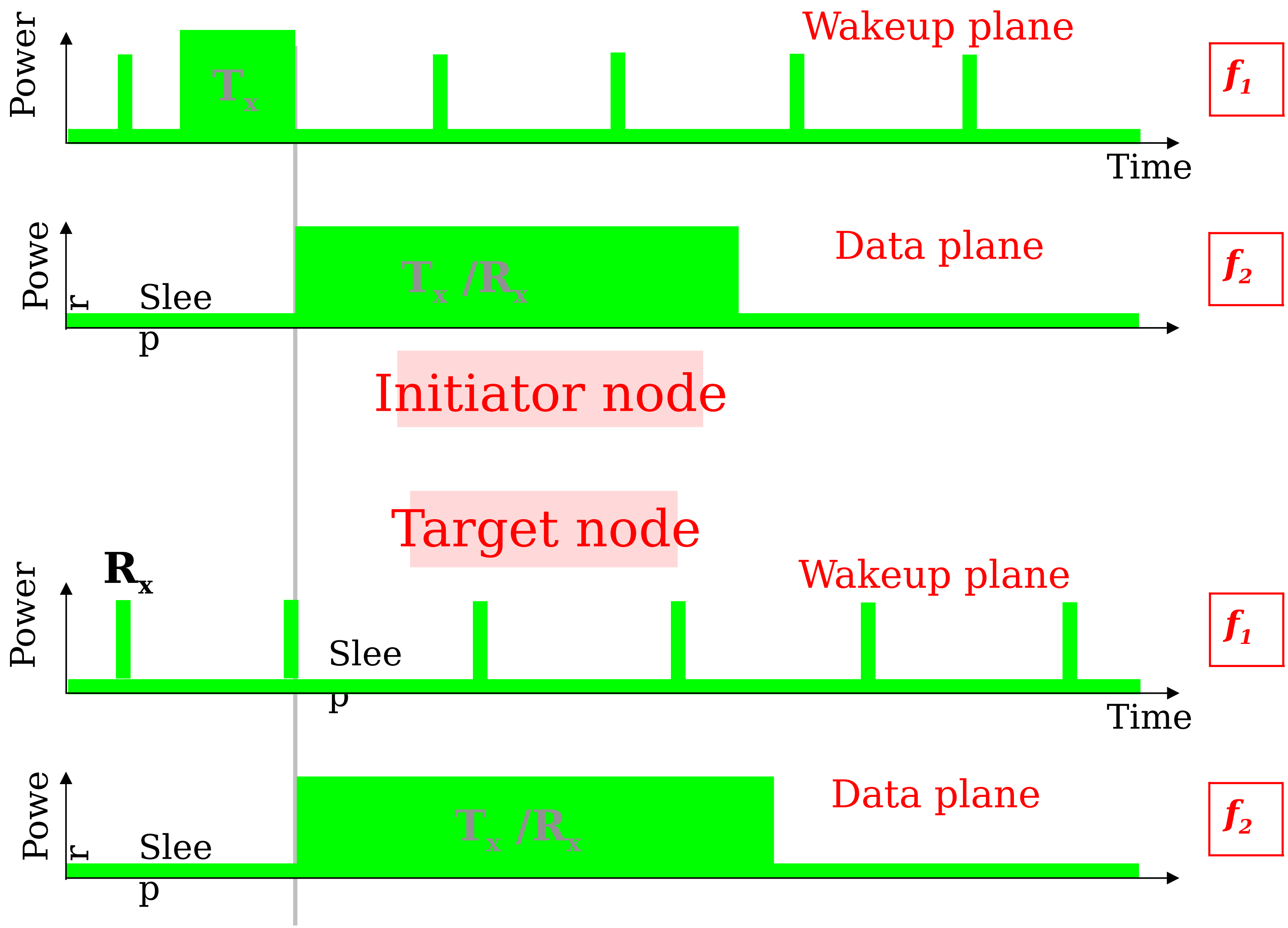
II. Topology Management

- **Two phases of sensor network operation**
  - Detect event
  - Relay information to users
- **Energy consumption of radio dominates that of sensors & CPU**

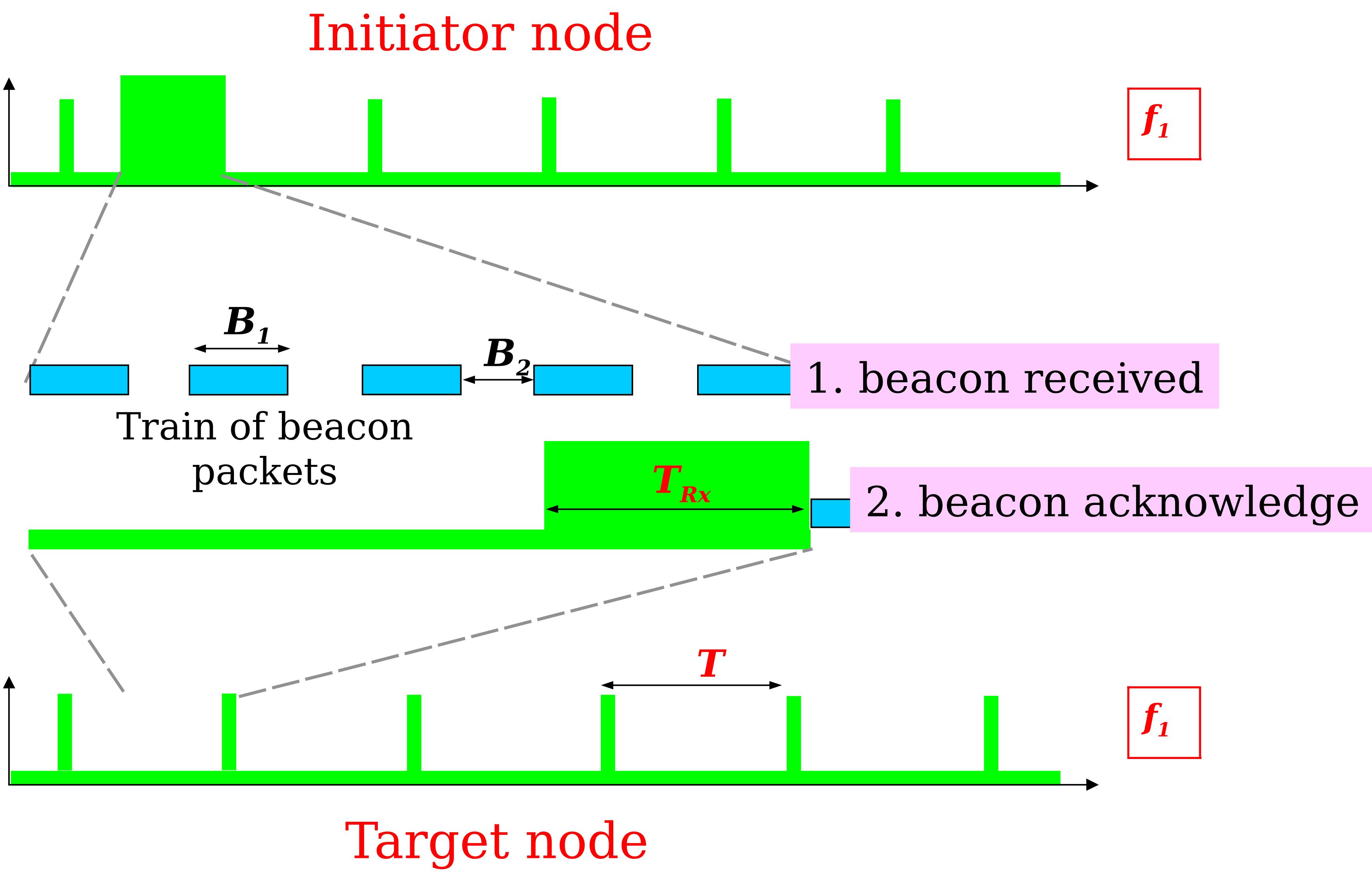
  ⇒ perform event detection continuously
- **The only energy efficient mode of the radio is the sleep mode**

  ⇒ put radio to sleep as often as possible
- **Existing approaches: density-energy trade-off**
  - keep enough nodes awake to handle the data forwarding (forwarding state)
  - but for substantial energy savings we need large densities
- **Observation**
  - most of the time, the network is only monitoring its environment, waiting for an event to happen (monitoring state)
- **Idea:**
  - put node radios to sleep and wake them up when they need to forward data
  - low duty-cycle paging channel using a 2nd radio: trades off energy savings for setup latency

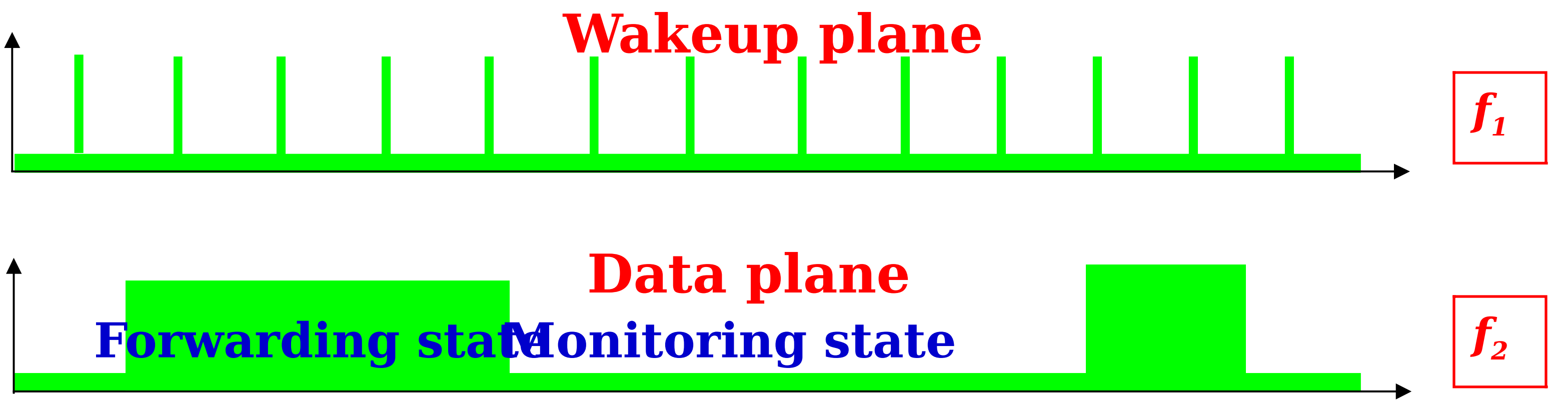
STEM: High-level Operation



Detailed Operation



Latency – Energy Analysis



Fraction of time in the forwarding state: $\boldsymbol{\alpha}$

$$\beta = 1/\alpha$$

- Setup latency

$$\bar{T}_S \approx \frac{T + \frac{2}{3}T_{Rx}}{2}$$

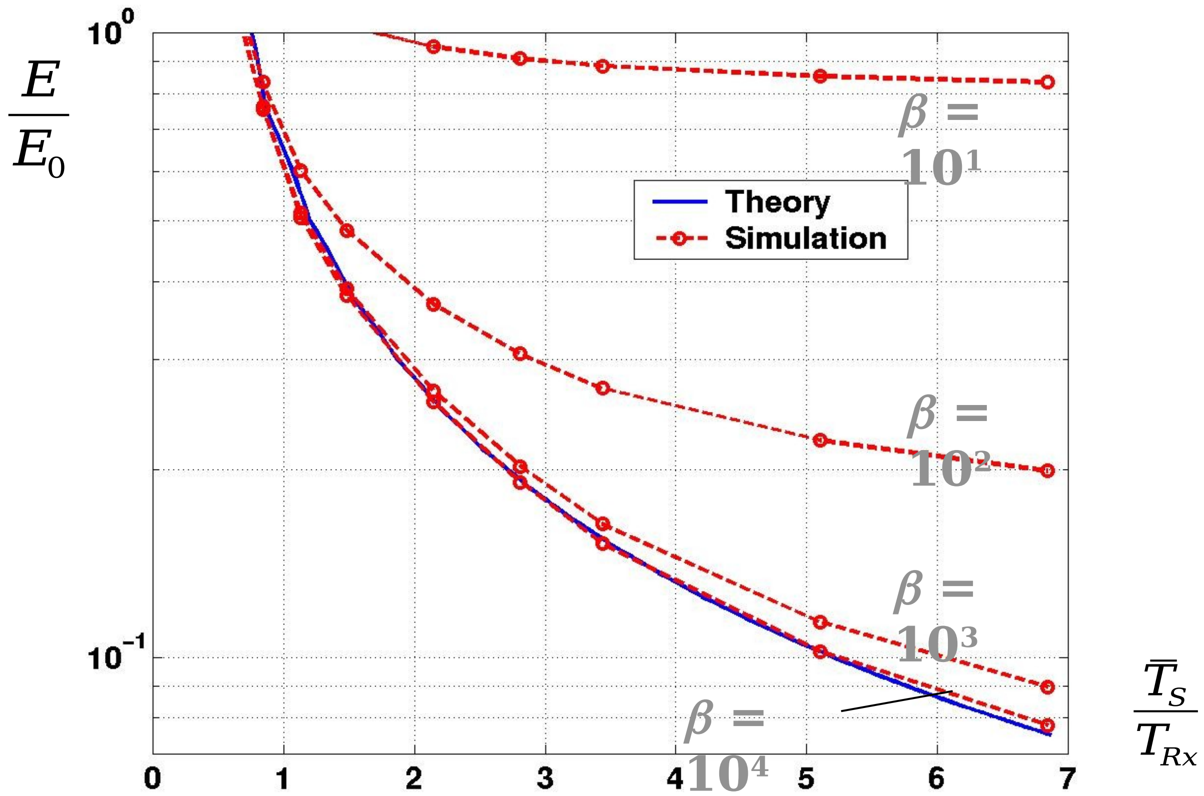
- Energy savings

$$\frac{E}{E_0} \approx \frac{T_{Rx}}{T}$$

Appropriate choice of interval sizes

Mostly monitoring state: $\alpha << 1$ or $\beta >> 1$

Energy-Latency Trade-off



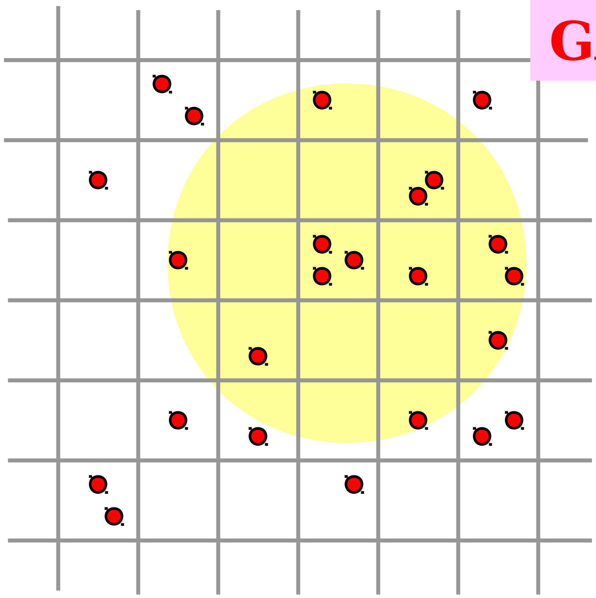
$T_{Rx} = 0.225$ s

- The tradeoff between energy and delay is manipulated by varying ***T***

$$T \uparrow \quad \Rightarrow \quad E \downarrow \quad T_S \uparrow$$

- The energy savings increase as the monitoring state becomes more dominant, $\beta \uparrow$

Topology Management in Forwarding State



GAF: Geographic Adaptive Fidelity [Ya2001]

- Conserve traffic forwarding capacity
- Divide network in virtual grids
- Each node in a grid is equivalent from a traffic forwarding perspective
- Keep **1 node awake in each grid** at e

- GAF reduces the energy by a factor ***M'***
- This factor is a function of the average number of nodes in a grid: ***M***

$$M' = \frac{M}{1 - e^{-M}}$$

for uniformly random node deployment

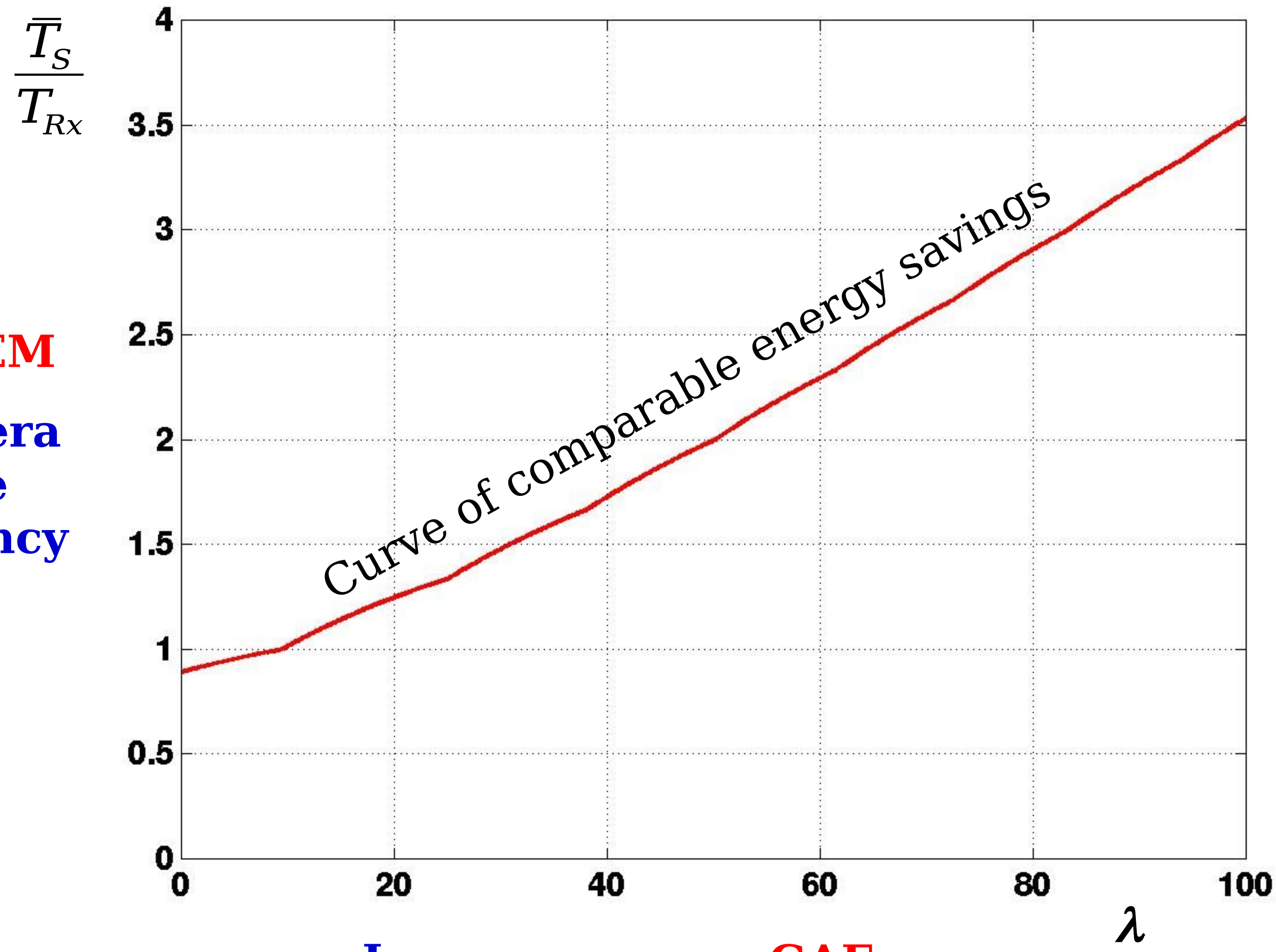
M'	M	λ
1.0	**0**	**0**
1.5	**0.87**	**13.7**
2.0	**1.59**	**25.0**
2.5	**2.22**	**35.0**
3.0	**2.82**	**44.3**

Average number of neighbors of a node

Comparing STEM & GAF

STEM

Leverage latency

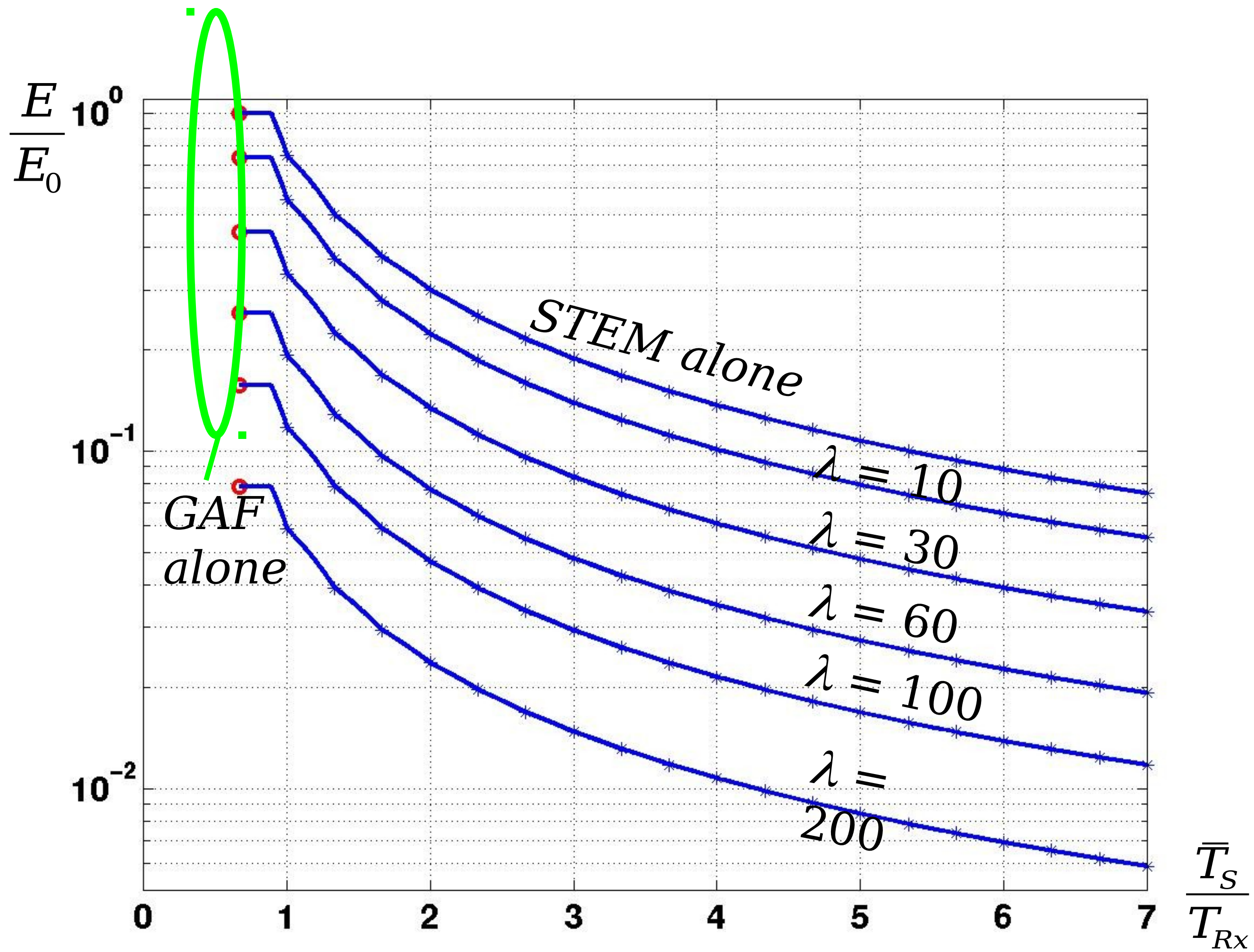


Leverage density

GAF

Combining STEM and GAF for Joint Energy-Latency-Density Trade-off

- As in GAF, 1 node is active in each grid
  ⇒ the grid can be considered a virtual node
- This virtual node runs the STEM protocol

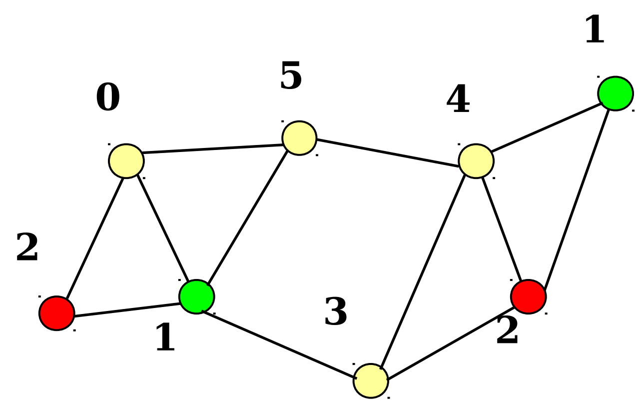


III. Dynamic MAC Address Assignment

- **Wireless spectrum is broadcast medium**
- **MAC addresses are required**

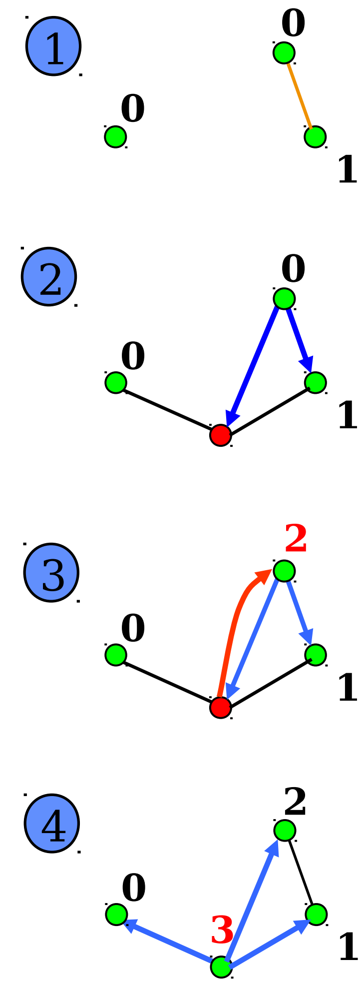
- **In wireless sensor networks, data size is small**
- **Unique MAC address present unneeded overhead**

- **Employ spatial address reuse (similar to reuse in cellular systems)**
  - **MAC address, link ids**
- **Two aspects**
  - Dynamic assignment algorithm
  - Address representation



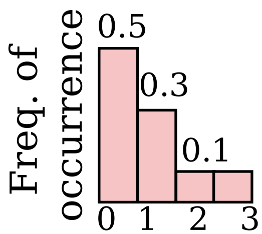
Distributed Assignment Algorithm

1. Network is operational (nodes have valid address)
2. Listen to periodic broadcasts of neighboring nodes
3. In case of conflict, notify node
   - (this node resends a broadcast)
4. Choose non-conflicting address and broadcast address in a periodic cycle. At this point the new node has joined the network.



- *Additive convergence:* network remains operational during address selection
- *Mapping:* unique ID to spatially reusable address
- Algorithm also valid when unidirectional links

Encoded Address Representation



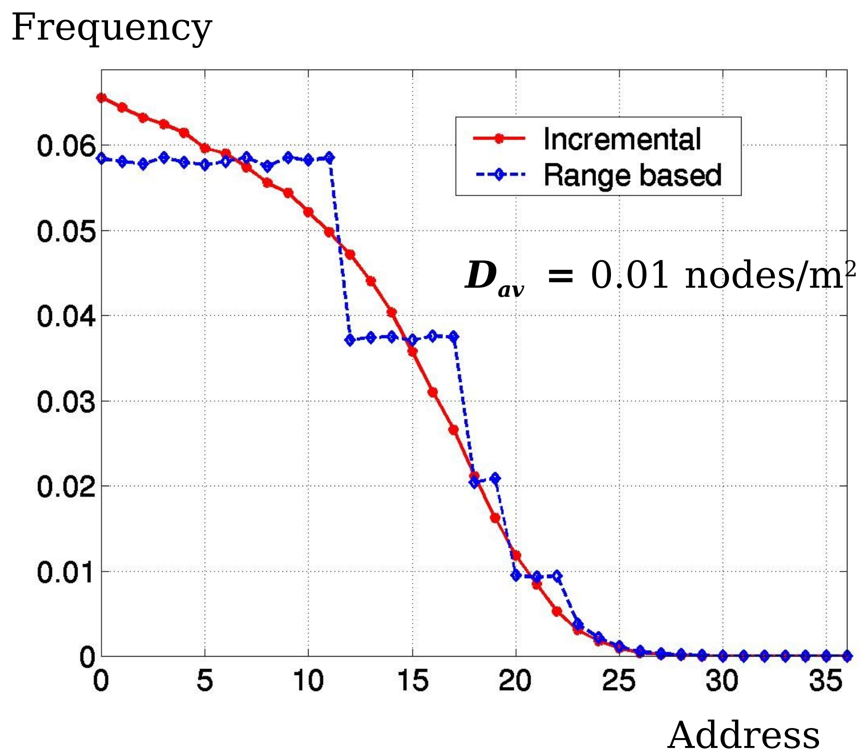
⇨

0	0
1	10
2	11
3	0 111

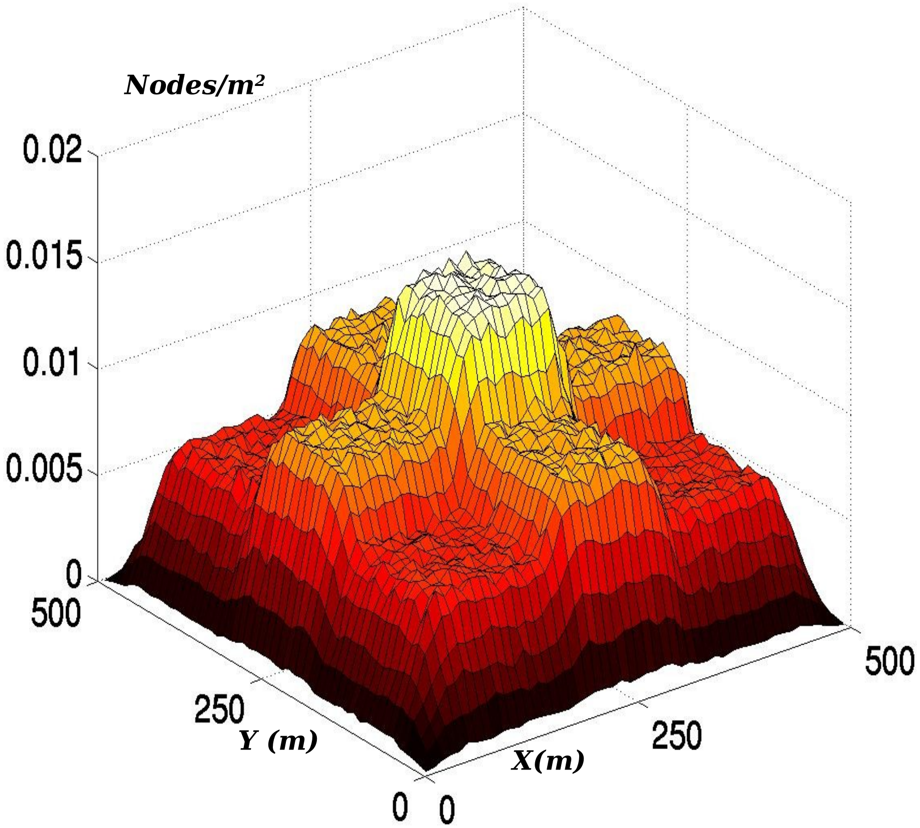
Encoded (bits/address)	**1.7**
Fixed size (bits/address)	**2**

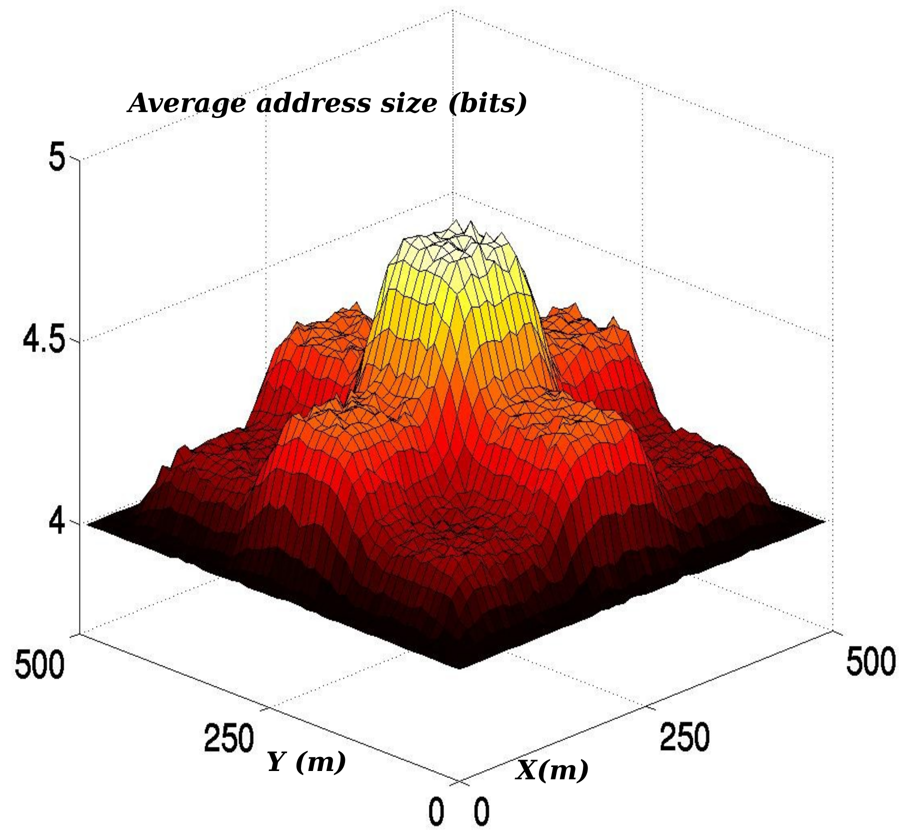
Address range	**0-11**	**12-17**	**18-19**	**20-22**	**23**	**…**
Codeword size (bits)	**4**	**5**	**6**	**7**	**8**	**…**

- **Size of the address field?**
- **Non-uniform address frequency**
  - **Huffman encoding**
  - **Robust: can represent any address**
- **Practical address selection**
  - **All addresses with same codeword size are equivalent**
  - **Choose random address in that range to reduce conflict messages**

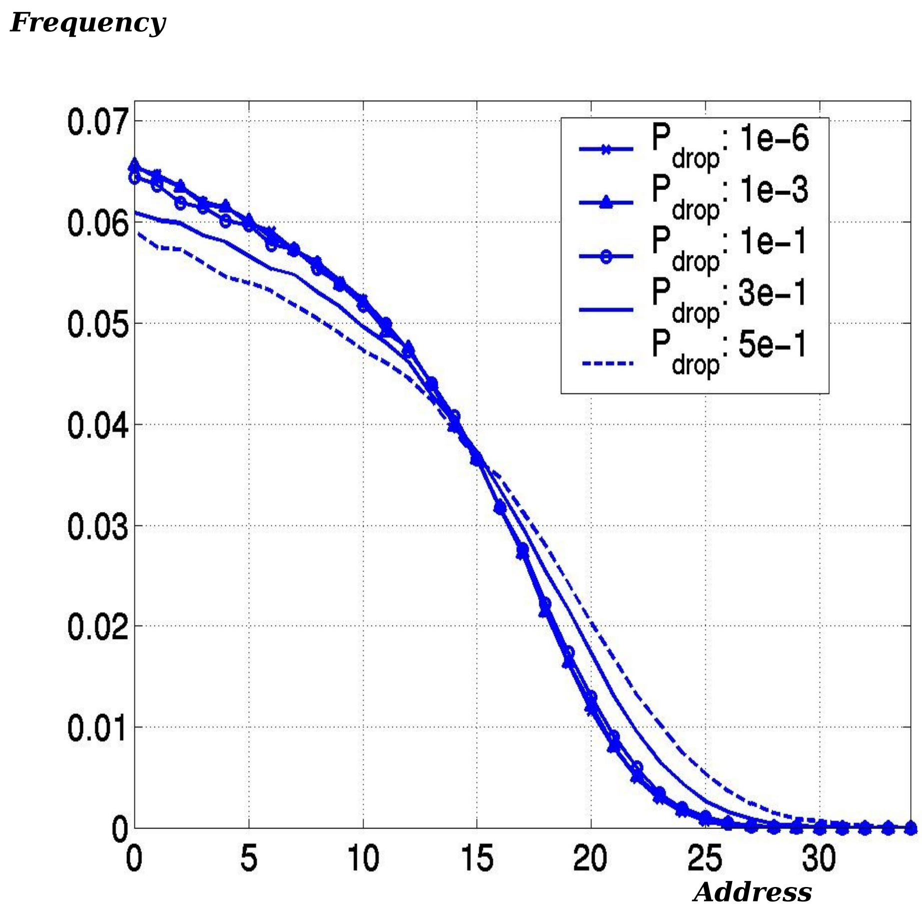


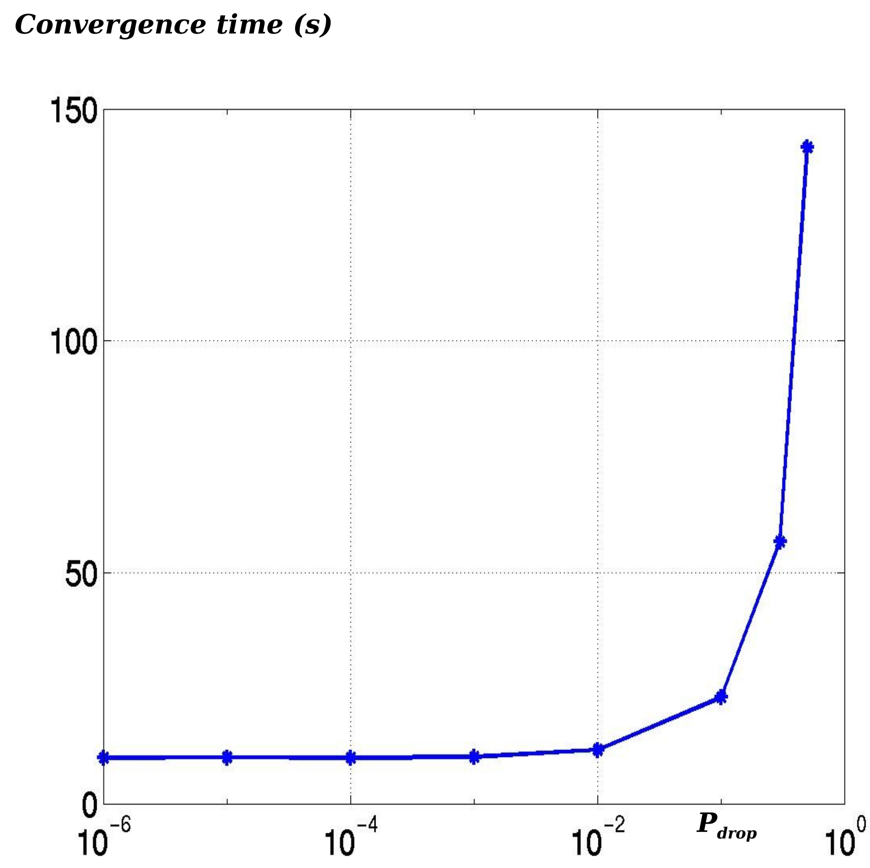
Non-uniform Network Density





Effect of Packet Losses (λ = 10)





Scalability

- **Address assignment**
  - **Distributed algorithm with periodic localized communication**
- **Address representation**
  - **Encoded addresses depend only on distribution**

Scales perfectly

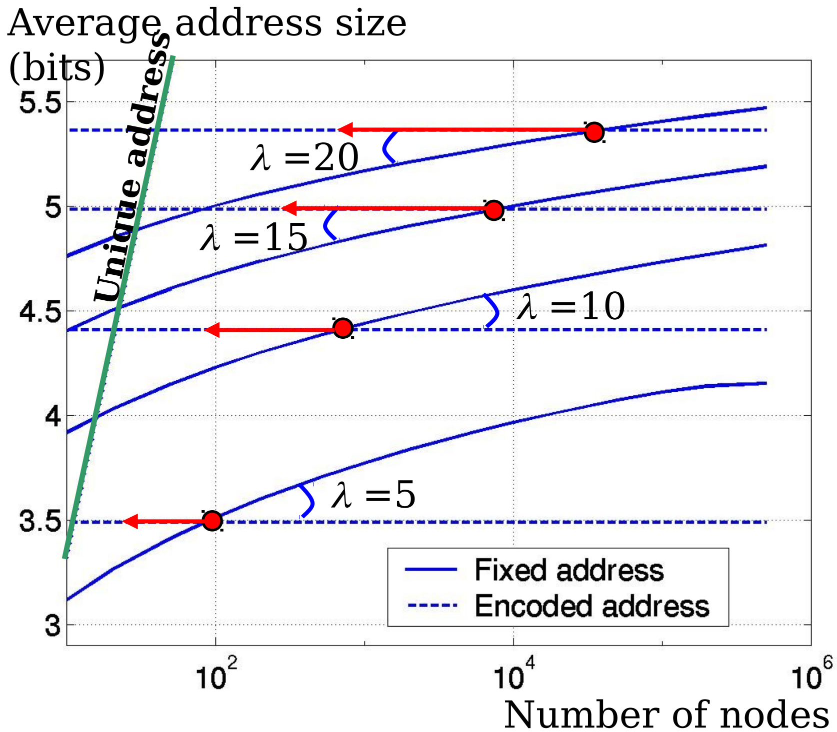
(neglecting edge effects)

Assignment

Off-line	**Centralized**	**Distributed**
++	-	+

Representation

Unique	**Fixed reusable**	**Encoded reusable**
--	$\pm$	+



Simulation Results

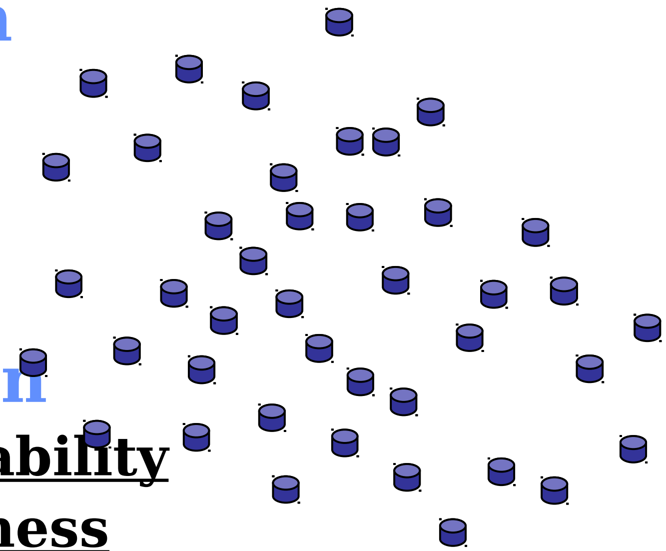
Scheme	Address selection type	Av. size (bits)	Address size scalability
Globally unique	Manufacturing	128	+
Network wide unique	Deployment	14	-
Fixed size dynamic	**Centr. / Distr.**	**4.7**	±
Encoded dynamic	**Distributed**	**4.4**	**+**

$\left.\begin{array}{l} \\ \end{array}\right\}$ Our schemes

Dynamic Address Allocation: Summary

- **Spatial reuse of address**

- **Dynamic assignment algorithm**
  - **Localized: <u>scalability</u>**
  - **Additive convergence: <u>robustness</u>**



- **Encoded address representation**
  - **Independent of network size: <u>scalability</u>**
  - **Variable length addresses: <u>robustness</u>**